

Summer 10-6-2011

A DYNAMIC LIFE-CYCLE MODEL FOR THE PROVISIONING OF SOFTWARE TESTING SERVICES: EXPERIENCES FROM A CASE STUDY IN THE CHINESE ICT SOURCING MARKET

Yikun Lu

Timo Käkölä

Follow this and additional works at: <http://aisel.aisnet.org/ecis2011>

Recommended Citation

Lu, Yikun and Käkölä, Timo, "A DYNAMIC LIFE-CYCLE MODEL FOR THE PROVISIONING OF SOFTWARE TESTING SERVICES: EXPERIENCES FROM A CASE STUDY IN THE CHINESE ICT SOURCING MARKET" (2011). *ECIS 2011 Proceedings*. 48.

<http://aisel.aisnet.org/ecis2011/48>

This material is brought to you by the European Conference on Information Systems (ECIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in ECIS 2011 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

A DYNAMIC LIFE-CYCLE MODEL FOR THE PROVISIONING OF SOFTWARE TESTING SERVICES: EXPERIENCES FROM A CASE STUDY IN THE CHINESE ICT SOURCING MARKET

Yikun Lu

University of Jyväskylä

yikun.y.lu@jyu.fi

Timo Käkölä

University of Jyväskylä

timokk@jyu.fi

Abstract

ICT-enabled international sourcing of software-intensive systems and services (eSourcing) is a powerful strategy for managing businesses more effectively. China is becoming a superpower for eSourcing service provisioning, but most Chinese providers are small or medium-sized and leverage the mediated eSourcing model, delivering services to foreign ICT clients that interface with end-clients onshore. This model restricts the providers to low-value projects. This paper probes eSourcing of software testing services within the Chinese market. Testing is studied for two reasons. First, testing is one of the best ICT services, small- and medium-sized providers can provide to develop domain and technological knowledge required to transcend the limitations of the mediated sourcing model. Second, high software quality is paramount as software has become ubiquitous. This paper draws upon a literature review and a case study to create and validate a life-cycle model for testing that helps providers develop the capabilities for overcoming the limitations of mediated sourcing. The class of integrated requirements, test, and defect management systems is found to be the most important class of information systems for enabling the life-cycle.

Keywords: eSourcing, Global software development, Software testing outsourcing, Requirements engineering and management.

1 Introduction

Based on the globalization of software business and the advances in information and communication technology (ICT), ICT-enabled sourcing of software-intensive systems and services (eSourcing) continues to increase. eSourcing can be domestic or international. International eSourcing refers to an ICT-enabled commercial arrangement, where an outsourcing service consumer commissions a foreign provider to provide software products or services formerly produced by the consumer. Domestic eSourcing takes place within one country.

Software development activities such as design, development, and testing as well as support and maintenance activities are eSourced extensively to offshore destinations such as India, China, and Russia (Adya, et al. 2008; Poston et al, 2010). Yet, eSourcing consumers need to deal with various challenges. Software developed by offshore providers does not always meet quality thresholds and/or functional requirements, projects are not delivered on time, too much management bandwidth is needed, and anticipated benefits such as cost savings are not received (Lacity and Rudramuniyaish, 2009; Lee, 2006; McCarthy, et al. 2004).

Many researchers have tried to find solutions to overcome the challenges and improve eSourcing performance. For example, Barthelemy (2006), Adya et al. (2008), and Dedrick et al. (2010) focused

on the process analysis, covering the eSourcing life-cycle from making sourcing decisions to project closure. Lacity and Rudramuniyaiah (2009) and Barthelemy (2006) focused, respectively, on cultural issues and crucial factors (e.g., the appropriateness of the selected provider) during the eSourcing processes. However, most extant research is based on the clients' perspective and does not give enough attention to the providers' viewpoints (Gonzalez et al, 2006). The literature does acknowledge that to achieve successful eSourcing, clients and providers need to work together to overcome cultural, communication, geographical, and other boundaries. Indeed, offshore services are critically dependent on a supply of providers capable of offering comparative cost advantage, satisfactory quality, and on time delivery despite the differences in distance, time zones and culture (Carmel and Tjia, 2005).

China has grown into one of the major sourcing service bases in the global ICT sourcing context (He, et al., 2008). However, most Chinese providers are small or medium-sized. They typically leverage the mediated offshore sourcing model, delivering software services to larger foreign ICT clients that contract and interface with the actual end-clients onshore (Järvenpää and Mao, 2008). This business model usually restricts the providers to small, low-value projects and hampers the sharing of knowledge with end-clients, severely impeding the capability and business development of Chinese providers. The extant literature does not extensively address this business model and ways to overcome its limitations. In addition, as the eSourcing of knowledge work has accelerated, theoretical models to explain the phenomenon have not kept up. For example, these models do not consider dynamic changes over time (Dedrick, Carmel and Kraemer, 2010).

Providers need comprehensive theoretical and practical guidance to overcome the restrictions of the mediated sourcing model. The research reported in this paper is part of a larger project to develop a generic, dynamic life-cycle model for the provisioning of ICT sourcing services that helps Chinese ICT service providers to develop dynamic capabilities for overcoming the limitations of the mediated sourcing model. The project will span three domains: ICT service sourcing (ICTS), business process sourcing (BPS) and knowledge process sourcing (KPS). Dynamic capabilities refer to the abilities of organizations to maintain their flexibility by creating competencies to address external pressures (Schwarz, Kalika, Kefi and Schwarz, 2010). They help clients and providers reconfigure human and other resources to address changing environments and requirements. Most importantly, they help providers to develop differentiated services and sometimes even high-tech products for international markets. We expect that providers and clients can draw upon the life-cycle model to establish dynamic capabilities, enabling them to interact transparently, monitor the life cycle in real time, identify communication and coordination breakdowns, and flexibly reconfigure resources to recover from breakdowns and to eliminate similar breakdowns proactively in future (c.f., Käkölä and Taalas, 2008).

This paper focuses on the ICTS domain and, especially, on the ICT-enabled sourcing of software testing services in the context of the Chinese ICT sourcing market. There are two reasons for this focus. First, software testing is one of the best ICT services small- and medium-sized Chinese providers can provide to develop in-depth domain and technological knowledge and other dynamic capabilities. Comprehensive testing services are practically impossible to develop without understanding the business domains of end-clients. Developing the domain knowledge takes time and effort but once providers have created it, they can deploy it, for example, to broaden the scope of services from testing to software product development, thus transcending the restrictions of the mediated sourcing model. Second, high software quality is paramount as software has become ubiquitous in society. Software testing is an empirical investigation conducted to provide stakeholders with information about the quality of the products and/or services under test, with respect to the context in which they are intended to operate (Kaner, 2006). It involves not only technical tasks but also considerations of economics and human psychology. Complete testing of complex applications takes too long and requires too many human resources to be economically feasible (Myers, 2004).

This paper draws upon a literature review and a case study in the context of the Chinese ICT sourcing market to create a dynamic eSourcing life-cycle model for the provisioning of software testing services. Dynamic capabilities are important for providers, because clients typically change their test requirements based on the results of test executions, requiring providers to replan the testing projects,

reconfigure resources, and adjust schedules. The investigated case organization has all the major characteristics of the Chinese ICT sourcing service providers. For example, it is medium-sized but it is growing fast and desires entry into the international sourcing market. It is thus an appropriate organization to start developing a generic life-cycle model for Chinese ICT sourcing service providers.

This paper focuses on the following research question: which software testing practices and information systems are the highest priority ones for Chinese testing service providers (hereafter, providers) from the viewpoint of executing the eSourcing life cycle, designing and delivering the services, recovering from coordination breakdowns, and proactively eliminating most breakdowns to improve service delivery and ensure organizational long-term effectiveness?

The paper proceeds as follows. The next section describes the research methodology and the case organization. Section “A dynamic eSourcing life-cycle model for software testing” presents the life-cycle model and validates it through a case study and a literature review. The section “Discussion” summarizes the most important components of the life-cycle model. Conclusions and ideas for future research are presented in the last section.

2 Description of the Research Methodology and the Case Organization

This paper draws upon a literature review and a longitudinal case study in the context of the Chinese ICT sourcing market to create a dynamic eSourcing life-cycle model for software testing providers. This research uses a single qualitative case study to collect data covering the complete life-cycle model, including the most important testing practices, the people responsible for specific testing assignments, and the information systems supporting the life-cycle.

The investigation has proceeded in the following stages. First, a reference model was selected from the literature to understand the international eSourcing life-cycle holistically from both clients’ and providers’ viewpoints. The eSourcing Capability Model for Service Providers (eSCM-SP) was chosen as the reference model because it has been demonstrated to help various types of providers to improve their capabilities related to both ongoing, phase-specific, and engagement-specific sourcing practices throughout the sourcing life-cycle (eSCM-SP, 2010). eSCM-SP can cover all the domains of this research project (ICTS, BPS and KPS). The model drafted in this paper has been aligned with eSCM-SP but it is more operational for testing service providers. The eSCM-SP life-cycle (Table 1) involves three phases from the provider’s viewpoint: initiation, delivery, and completion. Ongoing practices are run throughout the life-cycle to perform management functions. The three phases and the ongoing practices cover ten capability areas (e.g., knowledge management, threat management, performance management). The capability areas include 84 specific practices. eSCM-SP prescribes five capability levels. Certified assessors can use eSCM-SP to determine the capability levels of providers. Clients can use the certifications to find and select providers. Providers can use eSCM-SP as a roadmap to improve their capabilities to higher levels. This investigation collected data and compared the practices of the case organization to eSCM-SP based on the three phases and specific practices.

Second, scientific literature was reviewed in an iterative fashion to identify the key characteristics of successful testing providers (e.g., international growth orientation, sophisticated web-based integrated information systems). Third, Chinese software industry and software testing services industry were studied (Ma et al, 2008; Zhang et al, 2006; He et al, 2008) to identify the most suitable candidate for a case study. Fourth, a company called Ltesting was selected because it had the required characteristics.

Fifth, the first author spent over three weeks observing life in the case organization, analyzing documents and memoranda, and interviewing key personnel. In-depth interviews involved the CEO, all testing managers, and a number of test analysts to uncover the routine practices and information systems associated with testing work and the major breakdowns disrupting work. Each interview was started by following a questionnaire and concluded with an open discussion to address emerging

issues. Interviews were summarized and the summaries were sent to the interviewees, who verified them and provided feedback as necessary. Sixth, after the three-week visit, the data collected about testing strategies, routine practices, organizational structure, and enabling classes of information systems were analyzed to create the preliminary eSourcing life-cycle model. Most significant breakdowns in routines were also analyzed because the adoption of the finalized model should help organizations proactively eliminate most breakdowns. The phases of the preliminary model were compared to the respective phases prescribed by the eSCM-SP and to the relevant testing-related literature. If the analysis indicated that major deviations existed or information was missing, clarifications were requested from informants through email. Data collection and analysis continued for several months using the internet to collaborate with the case organization. A year after the first round of interviews, the first author performed a second round of interviews in the case company to collect supplementary data related to breakdowns and workarounds. This time, the quality assurance manager, the measurement process manager, and other people supporting the test teams were also interviewed. Due to space limitations, this paper focuses only on the work practices of the test teams.

Ongoing practices represent management functions that need to be performed during the entire eSourcing life-cycle in order to meet the intent of these practices.		
Initiation	Delivery	Completion
Practices focus on the capabilities needed to effectively prepare for service delivery. The practices are concerned with collecting and analyzing service requirements, negotiating, contracting, and designing and deploying the services, including the transfer of the necessary resources.	Practices focus on service delivery capabilities, including the ongoing management of service delivery, verification that commitments are being met, and the management of the finances associated with service provision.	Practices focus on the capabilities needed to effectively close an engagement with particular client(s) at the end of the eSourcing life-cycle. They include the capture of the lessons learned from the engagement and the transition of resources to the client, or to a third party, from the provider.

Table 1. The eSCM-SP V2.01

Title	Responsibility
Test Manager	Test managers are responsible for test project planning, management, risk evaluation, and report review. During project completion, test managers summarize the projects and the lessons learned. They have at least five years of relevant work experience.
Test Analyst	Test analysts analyze test requirements; design test plans (together with the test manager); and design test cases. They have three to four years of relevant work experience.
Tester	Testers conduct the specific test assignments. They have more than one year of work experience.
Seller	Sellers communicate with clients, acting as bridges between clients and providers. They need to have comprehensive testing knowledge, because they attend the testing service life-cycle from early bidding and negotiation through to service completion.

Table 1. Job Descriptions of Test Team Members

Ltesting is a medium-sized (less than 50 employees) professional software testing services provider founded in 2006 (Ltesting, 2010). Senior staff members (e.g., test managers and test analysts) have more than ten years of work experience in providing testing sourcing services. It has established a leading position in the Chinese testing service market and set up strategic partnerships with HP, IBM, and some other multinational companies. It offers software testing services, test training services for individuals and companies interested in offering testing services, test management services, and consulting services for constructing software quality systems. Ltesting expects the clients to be closely involved in the sourcing engagements in order to ensure the sourced projects meet clients' test requirements and help clients to obtain expected results. Ltesting implements most testing services offsite, being responsible solely for the test projects. It also offers onsite testing services: its testers join clients' test teams and are managed by the clients. To best address the research question probed in this paper, this research focuses on the projects following the offsite model.

Test teams are responsible for testing. Usually the teams have four roles: test manager, test analyst, tester, and seller (Table 2). Sellers serve as boundary spanners between clients and providers. They are especially important for solving communication challenges in international sourcing when clients and providers use different languages and have different cultures (Poston et al, 2010; Ma et al, 2008). Test teams can be organized flexibly based on the project characteristics, personnel workloads, and client requirements. For example, when the projects are small, testers need not be involved in test teams because test managers and test analysts can do their work.

3 A Dynamic eSourcing Life-cycle Model for Software Testing

This section presents the dynamic eSourcing life-cycle model for software testing from the provider's viewpoint (Figure 1). The relationships between clients and providers in the model are bilateral and dynamic as clients can modify test requirements and test plans during the life cycle. The model involves feedback loops, adjustments, and revisions over time, enabling providers and clients to communicate effectively, avoid misunderstandings, and quickly reconfigure resources (Beizer, 1990; Karinsalo and Abrahamsson, 2004; Ramler et al, 2005). To validate the model, this section investigates the case organization and analyzes the relevant testing practices to compare the provider's experiences to the reference model provided by eSCM-SP. The findings are organized based on the initiation, delivery, and completion phases of eSCM-SP to offer additional insights about the practices of the provider.

Ltesting uses HP Quality Center Software (QC, 2007) to manage the testing life cycle in collaboration with its clients. QC offers a web-based globally accessible suite of applications, supporting all essential aspects of testing from requirements management through test execution to defect management. Ltesting can serve its end-clients directly based on QC. There are many other similar suites in the market such as open source tools Bugfree and Testlink. This paper will use QC as an example to present how the requirements, test, and defect management tools in general support clients and providers during software testing projects.

3.1 Initiation

The initiation phase starts when a request for tender is received. The provider needs to assess the request and decide whether to create a tender or not. To make this decision, Ltesting's test manager and test analyst will usually work together to draft a preliminary test plan. If the plan shows that the tender represents a profitable opportunity, the tender is created and submitted to the client. If the contract is won, the client is responsible for providing the provider with the necessary resources and most importantly, the test requirements and access to people who created the requirements (e.g., requirements engineers, architects, and/or component developers). The provider develops a detailed test plan (together with the client) and the test cases and establishes the test environment.

3.1.1 Verify test requirements based on the request for tender

Clients present requests for tender to attract tenders and to evaluate and select the best providers. Requests for tender should include specific testing service requirements. Based on the request for tender, the provider analyzes test requirements and verifies the requirements. Test requirements are executable client requirements, which should fully cover client requirements, so no requirements are missed or overlooked. Sometimes, it is impossible to form an executable test plan because the test requirements are unclear or incomplete. If the applicable laws for tendering allow it, providers can request the client to elaborate on some requirements to ensure they understand what the client wants. Elaborations must be made available to all providers to afford fair bidding.

According to the case study and literature review (Boehm, 2001; Li, 2010), the biggest risk of testing eSourcing is service delay. Delays result if providers cannot verify the service scope and test require-

ments before test execution. Scoping is especially challenging when client requirements change often. Therefore, requirements management is crucial for clients and providers throughout the life cycle.

3.1.2 Form a brief test plan and bid

Providers draft preliminary test plans to assess the profitability of the requests for tenders and to show clients, how and why they can meet the test requirements. If the providers decide to bid, tests plans will be attached to the bids. Clients review and compare the bids and test plans with respect to their requests for tender, select the best providers, and sign contracts with the winning bidders. Ltesting's test plans include the estimated work effort (in person hours), time, and price.

"We need to assess the required work effort and bid for the project. Client will choose the most suitable provider based on their requirements. Price is only one of the considerations." (CEO)

To estimate the required work effort and form the preliminary test plan, Ltesting uses the Work Breakdown Structure (WBS) method. WBS is a tool used to define and group a project's discrete work elements (or tasks) in a way that helps define the total scope of the project (Brotheron et al, 2008). A complex project is broken down into individual components in a hierarchical structure, which can be outlined as a test task tree. Work elements can be completed independently of other elements, facilitating resource allocation, the assignment of responsibilities, and the monitoring of progress.

"Our previous experiences make estimates more accurate." (CEO)

3.1.3 Prepare and sign the contract

The contract should define responsibilities and commitments for the client and the provider (Gopal, et al. 2003). For example, the test scope and duration, the resources to be transferred or shared between the parties, security provisions, mechanisms for solving conflicts and dealing with service delays and requirements changes; rights for developed software assets and intellectual property, performance measures; quality assurance through process assessment methods such as ISO 15504 (SPICE) or CMMI, pricing; milestones and deliverables. Contracts help manage the relationships and control risks during the life cycle. If a client causes a service delay and the predefined work effort in hours is exceeded by more than 20 per cent, Ltesting will renegotiate the contract.

3.1.4 Form a detailed test plan

Ltesting drafts a detailed test plan and designs test cases after winning the contract. The plan includes the schedule and the methodologies and technologies to be used. Based on the WBS analysis, test managers define the test tasks. To outline a strategy to achieve the test requirements, they also define testing techniques (e.g., stress test, performance test), mechanisms to handle defects (e.g., severity classification, authorization to open and close defects), required resources (e.g., hardware, personnel), and milestones. Before Ltesting drafts a detailed test plan, it stores all the requirements documents received from clients (e.g., requirements specification and design specification) to a repository and lists them as a requirements tree, helping them manage test requirements during services.

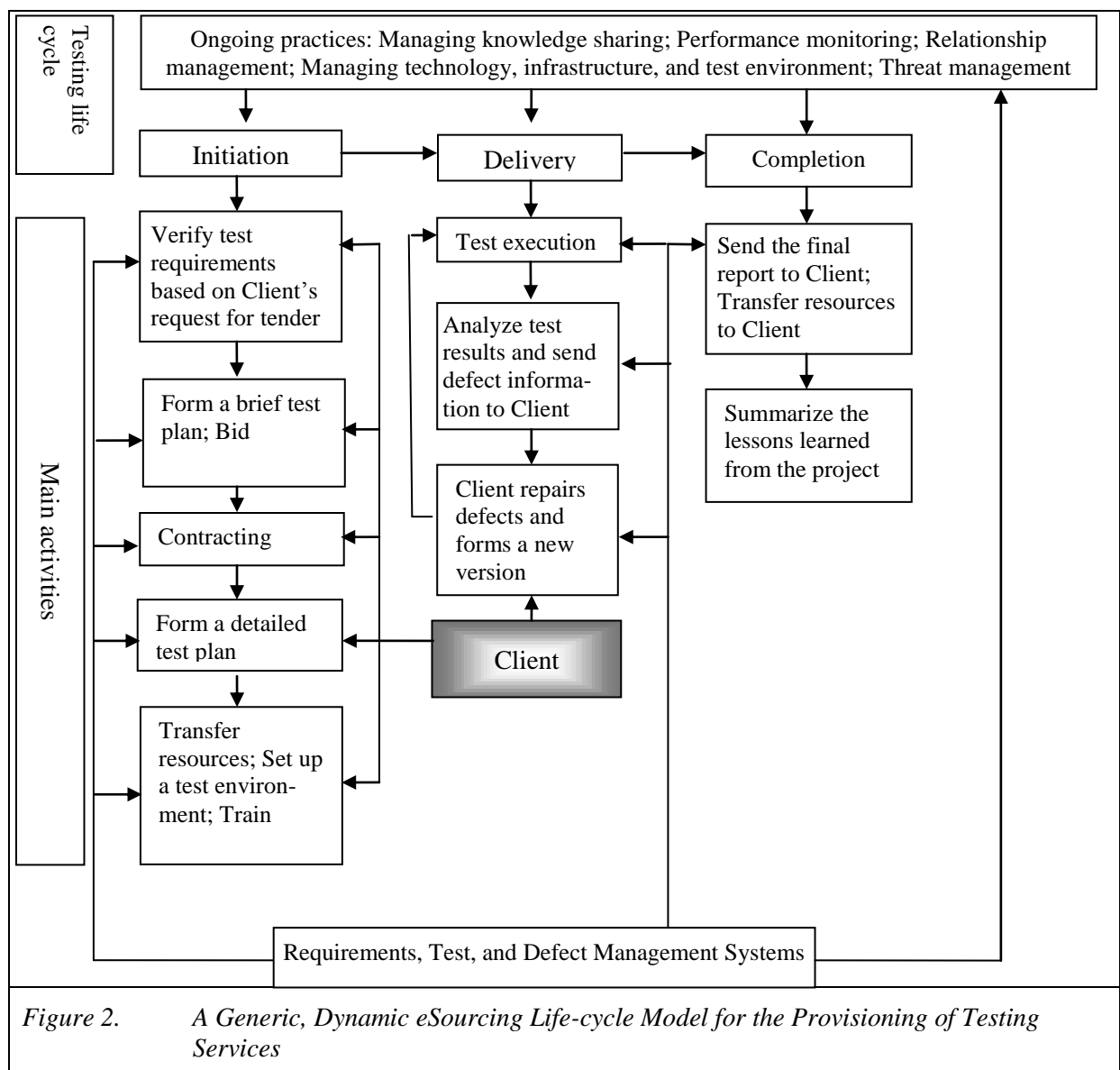
The client reviews the detailed test plan to determine how well it meets the goals defined in the beginning of the life cycle. When the client has approved the plan and test cases, the testing project will move to the delivery phase.

Ltesting can use QC to detail test plans in Microsoft Word documents. QC can read the plans from Microsoft Word documents and create the requirements trees and test plan trees accordingly. QC's requirements management module is an integrated solution for capturing, managing, and tracking requirements throughout the application development life cycle. Usually, Ltesting begins to use this module after it has detailed test plans. The extent to which Ltesting will use the module to capture requirements for the systems to be tested depends on the quality of clients' requirements specifications

and management processes. In the beginning of service engagements, it is impossible to know the maturity levels of the clients' requirements management processes. When maturity levels prove poor, Ltesting needs to communicate and negotiate with clients to elicit, clarify, and prioritize requirements and detail the scope of each engagement until both parties approve the scope.

"It depends on whether the client's requirements management and requirements specifications are poor or not. We seldom use QC to elicit requirements but we use QC to manage requirements."
(CEO)

When the service scope and test requirements have been validated test manager and test analyst will add traceability links between appropriate requirements. When clients change requirements during the delivery phase, traceability shows the other requirements the changes may affect.



3.1.5 Transfer resources, set up a test environment, and arrange training

Resources to be transferred or shared are elaborated in the detailed test plan. They include test equipment, infrastructure, software product or system to be tested, and the definition of the work context

where the system is to be used. Before starting the service delivery, the provider needs to set up a test environment and arrange training for the test team and the client representatives involved. During training, stakeholders share relevant business (domain), process, architectural, and organizational knowledge. Ltesting needs to familiarize itself with the test environments and hardware, whenever clients provide specific software and hardware platforms, such as high-performance servers.

In the eSCM-SP model, three capability areas are used in the initiation phase: contracting, service design and development, and service transfer. Ltesting uses all practices of these areas relevant to the initiation phase. In the completion phase, Ltesting uses all practices of the service transfer capability area relevant to the phase.

3.2 Delivery

This phase describes based on the case study and the literature review (Ammann and Offutt, 2008; QC, 2007) how providers usually deliver services based on test plans and how Ltesting implements requirements, test, and defect management using QC.

QC's release management module helps both clients and providers manage application releases and development cycles efficiently. The provider can track the progress of application development to determine whether the release will take place as planned and to make informed budgetary and release decisions accordingly. QC can be used to define different roles for clients and providers (e.g., developer, project manager, and tester). Each role has different access rights and authorities.

3.2.1 Test execution

Based on the detailed test plan and test requirements, the tester typically runs both automated and manual tests to find defects. Before test execution, Ltesting's test manager will allocate test tasks to testers and ensure all the test requirements are covered and traceable to tests. QC supports functional, regression, load, unit, integration, system and other types of testing. Each type of testing has its own set of requirements, schedules, and procedures. QC helps providers and clients to monitor and control the execution process.

3.2.2 Analyze the test results and send defect information to stakeholders

Locating application defects efficiently is the main purpose in the delivery phase. Following a test run, the provider analyzes the test results to identify which tests failed and which steps caused the failure. The analysis also needs to determine whether a defect has been detected in the application. If no defect caused the test failure, the expected results of the test may need to be updated.

When Ltesting's testers find defects in an application, they submit the defects to the respective QC project. The project stores defect information for retrieval by authorized users, such as members of the development, quality assurance, and support teams. To help clients repair the defects, the defect reports include detailed defect information such as related requirements, run steps, and related defects.

3.2.3 Client repairs defects and forms a new test version

Clients and providers need to work together effectively to manage the entire defect life-cycle, from initial problem detection through fixing the defect to verifying the fix. The provider sends information about the newly found defects to the client. The client's development team repairs the defects, submits a new release, and requests the provider to execute it and analyze the results. If some defects occur again or severe new defects are found, both parties need to continue for another round of the loop.

When requirements change, a change impact report details the affected requirements, enabling the provider to avoid a full regression test after each change. Regression testing can be performed

selectively or for the complete product. Normally, full regression testing is executed during the end of the testing cycle and partial regression testing is run between the test cycles (TestingGeek, 2010). QC notifies dispersed teams of any requirements changes possibly affecting the tasks they are working on.

To support asset sharing and reuse, QC provides version control for requirements, tests, test scripts, and business components. Versioning enables dispersed testing teams to manage multiple versions of test assets in parallel, while providing an audit trail of changes throughout the life cycle of each engagement. Version control thus helps clients and providers manage and track changes (Koivulahti-Ojala and Käkölä, 2010). The attributes of all the stored documents include name, status, version number, and author to help clients and providers avoid parallel, conflicting changes of the shared files in a multiuser environment.

The service delivery capability area of eSCM-SP includes eight practices. Ltesting uses all of them except for the practice “train clients” that is executed in the initiation phase.

3.3 Completion

During the completion phase, the provider prepares the final report, transfers resources to the client (and, possibly, to a third party), and summarizes the lessons learned from the project. Clients need to ensure that the results of the engagements meet predefined acceptance conditions (e.g., the defect curve is in the convergent state and all the requirements have been met).

3.3.1 Send the final test report

In the end of the testing life-cycle, the provider sends the final report to the client and transfers the resources agreed upon in the contract to the client or third parties. The final report should include the test results, the recorded defects, defect analyses, test logs, and other test documents. In addition to technology, infrastructure, and knowledge resources, the test cases are transferred to the client because, based on the industry convention, clients have the copyrights of test cases.

3.3.2 Summarize the lessons learned

Summarizing and documenting the lessons learned from the engagements is important for providers from the viewpoint of continuous improvement of service capabilities and quality. For example, Ltesting compares the actual service duration to the duration estimated in the test plan, the number of actual working hours to the estimated working hours, the testers’ actual performances to the expected ones, and the actual costs of resources to the estimated costs in the test plan. This information helps Ltesting to plan future projects more accurately and improve capabilities dynamically.

3.4 Ongoing practices

Ongoing practices represent management functions that need to be performed throughout the eSourcing life-cycle to meet the intent of these practices (eSCM-SP, 2010). The ongoing practices of eSCM-SP involve six capability areas. The practices of Ltesting include most of the practices of the six areas. However, Ltesting does not have appropriate practices to support innovation and continuously improve their service capabilities.

Most ongoing practices are enabled by requirements, test, and defect management systems during the testing life cycle. For example, the performance management capability area focuses on managing organizational performance so that the client requirements are met and the organization keeps learning and improving its performance. The area is enabled by an instance of such a class of systems (c.f., Käkölä, Koivulahti-Ojala, and Liimatainen, 2010). QC implements most requirements for the class in the case organization but it does not qualify as an instance of the class because other software products

and manual routines are necessary to enable the life-cycle. The case organization has not found a single solution to meet all its needs. Threat management and relationship management capability areas deal with the project and relationship risks. Poor project performance typically leads to relationship risks. Poor requirements management usually causes project risks such as service delays and breakdowns. Service breakdowns can happen at anytime and anyplace due to, for example, the changing client requirements. Ltesting has set up appropriate ongoing practices and supporting information systems (including QC) to proactively eliminate some breakdowns before they occur and to deal with the emerging breakdowns. When breakdowns occur, Ltesting typically creates new knowledge together with clients to resolve the situations and get routines back on track. Any changes in requirements will lead to the re-evaluation of the detailed test plan. Ltesting assesses the impacts of new, changed, and deleted requirements on the other requirements and the required work efforts mostly based on the traceability links between the requirements and between requirements and other test assets. Knowledge management capability area plays a crucial role in both sharing and securing critical knowledge assets and building trusted relationships. Ltesting creates generic test assets based on the test assets created in earlier engagements and adapts and reuses them in subsequent engagements, helping Ltesting to shorten the development time and improve the quality of test assets (e.g., test plans and test cases) and to achieve higher client satisfaction.

People management capability area refers to managing and motivating personnel to deliver services effectively. Based on the investigation, Ltesting has to improve its abilities in this capability area because existing competencies are not reviewed and developed systematically and career paths are not planned. Technology management capability area also needs to be improved. During each engagement, Ltesting arranges a person to manage the technology infrastructure with the client. However, no individual is specifically responsible for new technology initiatives such as researching and experimenting with innovations for automated software testing and test asset reuse. Additionally, clients often require Ltesting to deploy mature but costly technologies and methods for testing, imposing restrictions for Ltesting to innovate and improve its service abilities. In future, Ltesting needs to allocate more resources to carry out such initiatives on an ongoing basis to ensure it will remain a forerunner in its field.

4 Discussion

Clients' involvement and commitment to overcome the geographical, technological, cultural, and other sourcing barriers are critical to achieve successful software testing. While some software defects are caused by coding errors, the most expensive defects are caused by requirement gaps (e.g., unrecognized or misunderstood client requirements) (Kolawa and Huizinga, 2007). All the interviewees agreed that initiation is the most important phase in the testing life cycle and affects the other phases of the life cycle. Requirements analysis and test planning are conducted in this phase. If these activities fail, risks will materialize through breakdowns and service failures will result. In conclusion, initiation is the most important phase and requirements analysis and test planning are the most important practices for testing providers to control risks.

The interviewees indicated that information systems should enable a seamless and transparent testing life cycle from requirements elicitation, analysis, and prioritization through test planning, test case design, and execution to managing, repairing, and verifying defects. Therefore, the class of integrated requirements, test, and defect management systems is the most important class of information systems for testing. Instances of this class enable and are enabled by the effective execution of the eSourcing life-cycle model for testing providers, helping testing providers to meet change requirements quickly and improve service quality. The QC platform is an example of a commercial system supporting most common requirements for such an instance. However, the experiences from the case study, the literature review, and our earlier research (Käkölä et al., 2010) indicate that commercial and open source instances of the class successfully enabling the entire life-cycle are scarcely available.

The sourcing life-cycle model for testing services focuses on software testing but covers the relevant capability areas and practices of eSCM-SP. It is fully in line with the practices of Ltesting. Ltesting has been able to (1) deliver services directly to end clients through its transparent service life-cycle, (2) accumulate domain knowledge, and (3) communicate with all stakeholders effectively. As a result, it has successfully extended its service scope from the testing of banking software to financial and insurance services. Ltesting thus provides evidence that small and medium-sized software testing providers following the life-cycle model can overcome the limitations of the mediated sourcing model, for example, by extending the scope of their services to relevant domains and by enabling their clients to proactively communicate with them in order to deal with defects or change requirements.

5 Conclusions and Further Research

This research focused on the most important business practices and information systems for providers of software testing services to help providers tap the potential of global testing service provisioning markets. The paper created a comprehensive eSourcing life-cycle model for testing services, enabling providers and clients to manage the sourcing life cycle effectively. The paper validated the model through a literature review and a case study and by applying the model to evaluate the practices of the case organization. The extant literature does not present any similar models for testing eSourcing. The initiation phase proved most important in the testing eSourcing life cycle. It was also found that requirements analysis and test planning are the most important testing practices, primarily conducted in the initiation phase. The most important class of information systems for testing service providers is the class of requirements, test, and defect management systems.

The generalizability of this research was limited by the deployment of the single case study methodology in the Chinese context. The Quality Center suite used in the case organization may also have biased this research but the literature review and interviews indicated that the other commercially available tools are similar to QC. Future research has to investigate the practices and information systems of other testing providers and their clients. The case organizations need not use Quality Center but they may (and are likely to) use competing requirements, test, and/or defect management products. As a result, it is possible to create a novel information systems design theory for the class of requirements, test, and defect management systems (c.f., Käkölä et al., 2010). The theory helps clients and providers know what to expect from commercial product instances and to benchmark, select, and adopt products most suitable to their needs. The theory also helps software product providers and open source communities develop improved requirements, test, and defect management tools.

References

- Adya, M., Nath, D., Sridhar, V. and Malik, A. (2008). Bringing Global Sourcing into the Classroom: Lessons from an Experiential Software Development project, *Communications of the Association for Information Systems* (22), 33-48.
- Ammann, P. and Offutt, J. (2008). *Introduction to Software Testing*, Cambridge University Press.
- Barthelemy, J. (2006). The Seven Deadly Sins of Outsourcing, *Academy of Management Executive*, 17, 2, 87-98.
- Beizer, B. (1990). *Software Testing Techniques* (2nd edn), International Thomson Computer Press: New York NY.
- Boehm, B. and Basili, V.R. (2001). Software Defect Reduction Top 10 List, *IEEE Computer*, 34, 1, 135-137.
- Brotheron, S.A., Fried, R.T. and Norman, E.S. (2008). Applying the Work Breakdown Structure to the Project Management Lifecycle, *PMI Global Congress Proceedings*.
- Carmel, E. and Tjia, P. (2005). *Offshoring Information Technology: Sourcing and Outsourcing to a Global Workforce*, Cambridge University Press, Cambridge.

- Dedrick, J., Carmel, E. and Kraemer, K.L. (2010). A Dynamic Model of Offshore Software Development, *Journal of Information Technology*, 12, January.
- Gopal, K., Sivaramakrishnan, K., Krishnan, M.S., Mukhopadhyay, T. (2003). Contracts in Offshore Software Development: An Empirical Analysis, *Management Science*, 49, 12, 1671-1683.
- Gonzalez, R., Gasco, J. and Llopis, J. (2006). Information Systems Outsourcing: A Literature Analysis, *Information & Management*, 43, 821-834.
- He, M., Li, M. S., Wang, Q., Yang, Y. and Ye, K. (2008). An Investigation of Software Development Productivity in China, *ICSP' 08 Proceedings of the software process, 2008 international conference on making global distributed software development a success story*, 381-394.
- HP Quality Center Software Version (9.20) User's Guide (QC), (2007). Mercury Interactive Corporation, QCUG 9.2/01.
- Järvenpää, S. and Mao, J. (2008). Operational Capabilities Development in Mediated Offshore Software Services Models, *Journal of Information Technology* (23), 3-17.
- Käkölä, T., Koivulahti-Ojala, M. and Liimatainen, J. (2010). An Information Systems Design Product Theory for Integrated Requirements and Release Management Systems, *Journal of Software Maintenance and Evolution: Research and Practice*.
- Käkölä, T. and Taalas, A. (2008) Validating the Information Systems Design Theory for Dual Information Systems, *Proceedings of the 29th International Conference on Information Systems*, Paris, Association for Information Systems.
- Karinsalo, M. and Abrahamsson, P. (2004). Software Reuse and the Test Development Process: A Combined Approach, *International Conference on Software Reuse (ICSR)*, LNCS 3107, 59-68.
- Koivulahti-Ojala, M. and Käkölä, T. (2010). Framework for Evaluating the Version Management Capabilities of a Class of UML Modeling Tools from the Viewpoint of Multi-site, Multi-partner Product Line Organizations, *Proceedings of the 43rd Hawaii International Conference on Systems Sciences*, IEEE Computer Society, Hawaii, USA.
- Kolawa, A. and Huizinga, D. (2007). *Automated Defect Prevention: Best Practices in Software Management*, Wiley-IEEE Computer Society Press, 41-43, 86.
- Lacity, M. and Rudramuniyaiah, P.S. (2009). Funny Business: Public Opinion of Outsourcing and Offshoring as Reflected in U.S. and Indian Political Cartoons, *Communications of the Association for Information Systems*, 24, 13, 199-224.
- Lee, J.N. (2006). Outsourcing Alignment with Business Strategy and Firm Performance, *Communications of the Association for Information Systems*, 17, 49, 1124-1146.
- Li, Q., Yang, Y., Li, M.S., Wang, Q., Boehm, B. W. and Hu, C.Y. (2010). Improving Software Testing Process: Feature Prioritization to Make Winners of Success-critical Stakeholders, *Journal of Software Maintenance and Evolution: Research and Practice*.
- Ltesting (2010) <http://www.osourcing.net/>.
- Ma, J. Q., Li, J.Y., Chen, W.B., Conradi, R., Ji, J.Z. and Liu, C.N. (2008). A State of the Practice Study on Communication and Coordination Between Chinese Software Suppliers and Their Global Outsourcers, *Software Process Improvement and Practice*, 13, 3, 233-247.
- McCarthy, J.C., Martorelli, B., Moore, S., Agosta, L., and Ross, C.F. (2004). *Offshore Outsourcing: The Complete Guide*, Forrester Collection, September 7.
- Poston, R.S., Simon, J.C. and Jain, R. (2010). Client Communication Practices in Managing Relationships with Offshore Vendors of Software Testing Services, *Communications of the Association for Information Systems*, 27, 9, 129-148.
- Ramler, R., Biffl, S. and Grunbacher, P. (2005). *Value-based Management of Software Testing*, Value-based Software Engineering Springer: Berlin, 226-244.
- TestingGeek, Regression Testing, (2010). <http://www.testinggeek.com/index.php/testing-types/testing-purpose/104-regression-testing>.
- The eSourcing Capability Model for Service Providers (eSCM-SP) V2.01, 2010, <http://www.itsqc.org/index.html>.
- Zhang, S., Huang, L.H. and Yu, D.H. (2006). An Analysis of Information Systems Research in Chinese Mainland, *Communications of the Association for Information Systems*, 17, 35, 785-800.